

Reducing Higher Order π -Calculus to Spatial Logics

Zining Cao^{1,2,3}

¹Department of Computer Science and Technology
Nanjing University of Aero. & Astro.
Nanjing 210016, P. R. China

²Provincial Key Laboratory for Computer Information Processing Technology,
Soochow University
Suzhou 215006, P. R. China

³National Key Laboratory of Science and Technology on Avionics System Integration
Shanghai 200233, P. R. China
caozn@nuaa.edu.cn

Abstract. In this paper, we show that theory of processes can be reduced to the theory of spatial logic. Firstly, we propose a spatial logic SL for higher order π -calculus, and give an inference system of SL . The soundness and incompleteness of SL are proved. Furthermore, we show that the structure congruence relation and one-step transition relation can be described as the logical relation of SL formulae. We also extend bisimulations for processes to that for SL formulae. Then we extend all definitions and results of SL to a weak semantics version of SL , called WL . At last, we add μ -operator to SL . This new logic is named μSL . We show that WL is a sublogic of μSL and replication operator can be expressed in μSL .

1 Introduction

Higher order π -calculus was proposed and studied intensively in Sangiorgi's dissertation [29]. In higher order π -calculus, processes and abstractions over processes of arbitrarily high order, can be communicated. Some interesting equivalences for higher order π -calculus, such as barbed equivalence, context bisimulation and normal bisimulation, were presented in [29]. Barbed equivalence can be regarded as a uniform definition of bisimulation for a variety of concurrent calculi. Context bisimulation is a very intuitive definition of bisimulation for higher order π -calculus, but it is heavy to handle, due to the appearance of universal quantifications in its definition. In the definition of normal bisimulation, all universal quantifications disappeared, therefore normal bisimulation is a very economic characterization of bisimulation for higher order π -calculus. The coincidence between the three weak equivalences was proven [29,28,20]. Moreover, this proposition was generalized to strong case [10].

Spatial logic was presented in [12]. Spatial logic extends classical logic with connectives to reason about the structure of the processes. The additional connectives belong to two families. Intensional operators allow one to inspect the

structure of the process. A formula $A_1|A_2$ is satisfied whenever we can split the process into two parts satisfying the corresponding subformula A_i , $i = 1, 2$. In the presence of restriction in the underlying model, a process P satisfies formula $n\mathbb{R}A$ if we can write P as $(\nu n)P'$ with P' satisfying A . Finally, formula 0 is only satisfied by the inaction process. Connectives $|$ and \mathbb{R} come with adjunct operators, called guarantee (\triangleright) and hiding (\oslash) respectively, that allow one to extend the process being observed. In this sense, these can be called contextual operators. P satisfies $A_1 \triangleright A_2$ whenever the spatial composition (using $|$) of P with any process satisfying A_1 satisfies A_2 , and P satisfies $A \oslash n$ if $(\nu n)P$ satisfies A . Some spatial logics have an operator for fresh name quantification [11].

There are lots of works of spatial logics for π -calculus and *Mobile Ambients*. In some papers, spatial logic was studied on its relations with structural congruence, bisimulation, model checking and type system of process calculi [5,6,9,16,27]. The main idea of this paper is that the theory of processes can be reduced to the theory of spatial logic. In this paper, we present a spatial logic for higher order π -calculus, called *SL*, which comprises some action temporal operators such as $\langle \tau \rangle$ and $\langle a \langle A \rangle \rangle$, some spatial operators such as prefix and composition, some adjunct operators of spatial operators such as \triangleright and \oslash , and some operators on the property of free names and bound names such as $\ominus n$ and $\tilde{\ominus}$. We give an inference system of *SL*, and prove the soundness of the inference system for *SL*. Furthermore, we show that there is no finite complete inference system for *SL*. Then we study the relation between processes and *SL* formulas. We show that a *SL* formula can be viewed as a specification of processes, and conversely, a process can be viewed as a special kind of *SL* formulas. Therefore, *SL* is a generalization of processes, which extend process with specification statements. We show that the structure congruence relation and one-step transition relation can be described as the logical relation of *SL* formulas. We also show that bisimulations for higher order processes can be characterized by a sublogic of *SL*. Furthermore, we give a weak semantics version of *SL*, called *WL*, where the internal action is unobservable. The results of *SL* are extended to *WL*, such as an inference system for *WL*, the soundness of this inference system, and no finite complete inference system for *WL*. Finally, we add μ -operator to *SL*. The new logic named μSL . We show that *WL* is a sublogic of μSL and replication operator can be expressed in μSL . Thus μSL is a powerful logic which can express both strong semantics and weak semantics for higher order processes.

This paper is organized as follows: In Section 2, we briefly review higher order π -calculus. In Section 3, we present a spatial logic *SL*, including its syntax, semantics and inference system. The soundness and incompleteness of inference system of *SL* are proved. Furthermore, we discuss that *SL* can be regarded as a specification language of processes and processes can be regarded as a kind of special formulas of *SL*. Bisimulation in higher order π -calculus is described by a sublogic of *SL*. In Section 4, we give a weak semantics version of *SL*, called *WL*. We generalize concepts and results of *SL* to *WL*. In Section 5, we add μ -operator to *SL*. The new logic named μSL . The expressive power of μSL is studied. The paper is concluded in Section 6.

2 Higher Order π -Calculus

2.1 Syntax and Labelled Transition System

In this section we briefly recall the syntax and labelled transition system of the higher order π -calculus. Similar to [28], we only focus on a second-order fragment of the higher order π -calculus, i.e., there is no abstraction in this fragment.

We assume a set N of names, ranged over by a, b, c, \dots and a set Var of process variables, ranged over by X, Y, Z, U, \dots . We use E, F, P, Q, \dots to stand for processes. Pr denotes the set of all processes.

We first give the grammar for the higher order π -calculus processes as follows:

$$P ::= 0 \mid U \mid \pi.P \mid P_1 \mid P_2 \mid (\nu a)P$$

π is called a prefix and can have one of the following forms:

$\pi ::= a(U) \mid \bar{a}\langle P \rangle$, here $a(U)$ is a higher order input prefix and $\bar{a}\langle P \rangle$ is a higher order output prefix.

In each process of the form $(\nu y)P$ the occurrence of y is bound within the scope of P . An occurrence of y in a process is said to be free iff it does not lie within the scope of a bound occurrence of y . The set of names occurring free in P is denoted $fn(P)$. An occurrence of a name in a process is said to be bound if it is not free, we write the set of bound names as $bn(P)$. $n(P)$ denotes the set of names of P , i.e., $n(P) = fn(P) \cup bn(P)$. The definition of substitution in process terms may involve renaming of bound names when necessary to avoid name capture.

Higher order input prefix $a(U).P$ binds all free occurrences of U in P . The set of variables occurring free in P is denoted $fv(P)$. We write the set of bound variables as $bv(P)$. A process is closed if it has no free variable; it is open if it may have free variables. Pr^c is the set of all closed processes.

Processes P and Q are α -convertible, $P \equiv_\alpha Q$, if Q can be obtained from P by a finite number of changes of bound names and variables. For example, $(\nu b)(\bar{a}\langle b(U).U \rangle.0) \equiv_\alpha (\nu c)(\bar{a}\langle c(U).U \rangle.0)$.

Structural congruence: $P \mid Q \equiv Q \mid P$; $(P \mid Q) \mid R \equiv P \mid (Q \mid R)$; $P \mid 0 \equiv P$; $(\nu a)0 \equiv 0$; $(\nu m)(\nu n)P \equiv (\nu n)(\nu m)P$; $(\nu a)(P \mid Q) \equiv P \mid (\nu a)Q$ if $a \notin fn(P)$.

In [26], Parrow has shown that in higher order π -calculus, the replication can be defined by other operators such as higher order prefix, parallel and restriction. For example, $!P$ can be simulated by $R_P = (\nu a)(D \mid \bar{a}\langle P \mid D \rangle.0)$, here $D = a(X).(X \mid \bar{a}\langle X \rangle.0)$.

The operational semantics of higher order processes is given in Table 1. We have omitted the symmetric cases of the parallelism and communication rules.

$$\begin{aligned} ALP : & \frac{P \xrightarrow{\alpha} P'}{Q \xrightarrow{\alpha} Q'} P \equiv Q, P' \equiv Q' \\ OUT : & \bar{a}\langle E \rangle.P \xrightarrow{\bar{a}\langle E \rangle} P \\ IN : & a(U).P \xrightarrow{a\langle E \rangle} P\{E/U\} \text{ } bn(E) = \emptyset \\ PAR : & \frac{P \xrightarrow{\alpha} P'}{P \mid Q \xrightarrow{\alpha} P' \mid Q} bn(\alpha) \cap fn(Q) = \emptyset \end{aligned}$$

$$\begin{aligned}
COM : & \frac{P \xrightarrow{(\nu \tilde{b})\bar{a}\langle E \rangle} P' \quad Q \xrightarrow{a\langle E \rangle} Q'}{P|Q \xrightarrow{\tau} (\nu \tilde{b})(P'|Q')} \quad \tilde{b} \cap fn(Q) = \emptyset \\
RES : & \frac{P \xrightarrow{\alpha} P'}{(\nu a)P \xrightarrow{\alpha} (\nu a)P'} \quad a \notin n(\alpha) \\
OPEN : & \frac{P \xrightarrow{(\nu \tilde{c})\bar{a}\langle E \rangle} P'}{(\nu b)P \xrightarrow{(\nu b, \tilde{c})\bar{a}\langle E \rangle} P'} \quad a \neq b, \quad b \in fn(E) - \tilde{c}
\end{aligned}$$

Table 1

2.2 Bisimulations in Higher Order π -Calculus

Context bisimulation and contextual barbed bisimulation were presented in [29,28] to describe the behavioral equivalences for higher order π -calculus. Let us review the definition of these bisimulations. In the following, we abbreviate $P\{E/U\}$ as $P\langle E \rangle$.

Context bisimulation is an intuitive definition of bisimulation for higher order π -calculus.

Definition 1 A symmetric relation $R \subseteq Pr^c \times Pr^c$ is a strong context bisimulation if $P R Q$ implies:

- (1) whenever $P \xrightarrow{\tau} P'$, there exists Q' such that $Q \xrightarrow{\tau} Q'$ and $P' R Q'$;
- (3) whenever $P \xrightarrow{a\langle E \rangle} P'$, there exists Q' such that $Q \xrightarrow{a\langle E \rangle} Q'$ and $P' R Q'$;
- (4) whenever $P \xrightarrow{(\nu \tilde{b})\bar{a}\langle E \rangle} P'$, there exist Q', F, \tilde{c} such that $Q \xrightarrow{(\nu \tilde{c})\bar{a}\langle F \rangle} Q'$ and for all $C(U)$ with $fn(C(U)) \cap \{\tilde{b}, \tilde{c}\} = \emptyset$, $(\nu \tilde{b})(P'|C\langle E \rangle) R (\nu \tilde{c})(Q'|C\langle F \rangle)$. Here $C(U)$ represents a process containing a unique free variable U .

We write $P \sim_{Ct} Q$ if P and Q are strongly context bisimilar.

Contextual barbed equivalence can be regarded as a uniform definition of bisimulation for a variety of process calculi.

Definition 2 A symmetric relation $R \subseteq Pr^c \times Pr^c$ is a strong contextual barbed bisimulation if $P R Q$ implies:

- (1) $P|C R Q|C$ for any C ;
- (2) whenever $P \xrightarrow{\tau} P'$ then there exists Q' such that $Q \xrightarrow{\tau} Q'$ and $P' R Q'$;
- (3) $P \downarrow_\mu$ implies $Q \downarrow_\mu$, where $P \downarrow_a$ if $\exists P', P \xrightarrow{a\langle E \rangle} P'$, and $P \downarrow_{\bar{a}}$ if $\exists P', P \xrightarrow{(\nu \tilde{b})\bar{a}\langle E \rangle} P'$.

We write $P \sim_{Ba} Q$ if P and Q are strongly contextual barbed bisimilar.

Intuitively, tau action represents the internal action of processes. If we just consider external actions, then we should adopt weak bisimulations to characterize the equivalence of processes.

Definition 3 A symmetric relation $R \subseteq Pr^c \times Pr^c$ is a weak context bisimulation if $P R Q$ implies:

- (1) whenever $P \xRightarrow{\varepsilon} P'$, there exists Q' such that $Q \xRightarrow{\varepsilon} Q'$ and $P' R Q'$;
- (2) whenever $P \xrightarrow{a\langle E \rangle} P'$, there exists Q' such that $Q \xrightarrow{a\langle E \rangle} Q'$ and $P' R Q'$;

(3) whenever $P \xrightarrow{(\nu\tilde{b})\tilde{a}\langle E \rangle} P'$, there exist Q', F, \tilde{c} such that $Q \xrightarrow{(\nu\tilde{c})\tilde{a}\langle F \rangle} Q'$ and for all $C(U)$ with $fn(C(U)) \cap \{\tilde{b}, \tilde{c}\} = \emptyset$, $(\nu\tilde{b})(P'|C\langle E \rangle) R (\nu\tilde{c})(Q'|C\langle F \rangle)$. Here $C(U)$ represents a process containing a unique free variable U .

We write $P \approx_{Ct} Q$ if P and Q are weakly context bisimilar.

Definition 4 A symmetric relation $R \subseteq Pr^c \times Pr^c$ is a weak contextual barbed bisimulation if $P R Q$ implies:

- (1) $P|C R Q|C$ for any C ;
- (2) whenever $P \xRightarrow{\varepsilon} P'$ then there exists Q' such that $Q \xRightarrow{\varepsilon} Q'$ and $P' R Q'$;
- (3) $P \Downarrow_\mu$ implies $Q \Downarrow_\mu$, where $P \Downarrow_\mu$ if $\exists P', P \xRightarrow{\varepsilon} P'$ and $P' \Downarrow_\mu$.

We write $P \approx_{Ba} Q$ if P and Q are weakly contextual barbed bisimilar.

3 Logics for Strong Semantics

In this section, we present a logic to reason about higher order π -calculus called *SL*. This logic extends propositional logic with three kinds of connectives: action temporal operators, spatial operators, operators about names and variables. We give the syntax and semantics of *SL*. The inference system of *SL* is also given. We prove the soundness and incompleteness of this inference system. As far as we know, this is the first result on the completeness problem of the inference system of spatial logic. Furthermore, we show that structural congruence, one-step transition relation and bisimulation can all be characterized by this spatial logic. It is well known that structural congruence, one-step transition relation and bisimulation are the central concept in the theory of processes, and almost all the studies of process calculi are about these concepts. Therefore, our study gives an approach of reducing theory of processes to theory of spatial logic. Moreover, since processes can be regarded as a special kind of spatial logic formulas, spatial logic can be viewed as an extension of process calculus. Based on spatial logic, it is possible to propose a refinement calculus [23] of concurrent processes.

3.1 Syntax and Semantics of Logic *SL*

Now we introduce a logic called *SL*, which is a spatial logic for higher order π -calculus.

Definition 5 Syntax of logic *SL*

$$A ::= \top \mid \perp \mid \neg A \mid A_1 \wedge A_2 \mid \langle \tau \rangle A \mid \langle a \langle A_1 \rangle \rangle A_2 \mid \langle a[A_1] \rangle A_2 \mid \langle \bar{a} \langle A_1 \rangle \rangle A_2 \mid 0 \mid X \mid a \odot X.A \mid A \setminus a \odot X \mid \bar{a} \langle A_1 \rangle . A_2 \mid A \setminus \bar{a} \mid A_1 | A_2 \mid A_1 \triangleright A_2 \mid a \textcircled{R} A \mid A \odot a \mid (\mathbf{N}x)A \mid (\mathbf{N}X)A \mid (\ominus a)A \mid (\tilde{\ominus})A \mid a \neq b$$

In $(\mathbf{N}x)A$, $(\mathbf{N}X)A$, the variables x (and X) are bound with scope the formula A . We assume defined on formulas the standard relation \equiv_α of α -conversion (safe renaming of bound variables), but we never implicitly take formulas “up to α -conversion”: our manipulation of variables via α -conversion steps is always quite explicit. The set $fn(A)$ of free names in A , and the set $fpv(A)$ of free propositional variables in A , are defined in the usual way. A formula is closed if

it has no free variable such as X , it is open if it may have free variables. SL^c is the set of all closed formulas. In the following, we use $A\{b/a\}$ to denote the formula obtained by replacing all occurrence of a in A by b . Similarly, we use $A\{Y/X\}$ to denote the formula obtained by replacing all occurrence of Y in A by X . It is easy to see that a process can also be regarded as a spatial logic. For example, process $\bar{a}\langle E \rangle.P$ is also a spatial logic. In this paper, we say that such a formula is in the form of process.

Definition 6 Semantics of logic SL

$$\begin{aligned}
[[\top]]_{Pr} &= Pr \\
[[\perp]]_{Pr} &= \emptyset \\
[[\neg A]]_{Pr} &= Pr - [[A]]_{Pr} \\
[[A_1 \wedge A_2]]_{Pr} &= [[A_1]]_{Pr} \cap [[A_2]]_{Pr} \\
[[\langle \tau \rangle A]]_{Pr} &= \{P \mid \exists Q. P \xrightarrow{\tau} Q \text{ and } Q \in [[A]]_{Pr}\} \\
[[\langle a \langle A_1 \rangle \rangle A_2]]_{Pr} &= \{P \mid \exists P_1, P_2. P \xrightarrow{a \langle P_1 \rangle} P_2, P_1 \in [[A_1]]_{Pr} \text{ and } P_2 \in [[A_2]]_{Pr}\} \\
[[\langle a \langle A_1 \rangle \rangle A_2]]_{Pr} &= \{P \mid \forall R, R \in [[A_1]]_{Pr}, \exists Q. P \xrightarrow{a \langle R \rangle} Q \text{ and } Q \in [[A_2]]_{Pr}\} \\
[[\langle \bar{a} \langle A_1 \rangle \rangle A_2]]_{Pr} &= \{P \mid \exists P_1, P_2. P \xrightarrow{(\nu \tilde{b}) \bar{a} \langle P_1 \rangle} P_2, (\nu \tilde{b}) P_1 \in [[A_1]]_{Pr} \text{ and } P_2 \in [[A_2]]_{Pr}\} \\
[[0]]_{Pr} &= \{P \mid P \equiv 0\} \\
[[X]]_{Pr} &= \{P \mid P \equiv X\} \\
[[a \odot X.A]]_{Pr} &= \{P \mid \exists Q. P \equiv a(X).Q \text{ and } Q \in [[A]]_{Pr}\} \\
[[A \setminus a \odot X]]_{Pr} &= \{P \mid a(X).P \in [[A]]_{Pr}\} \\
[[\bar{a} \langle A_1 \rangle . A_2]]_{Pr} &= \{P \mid \exists P_1, P_2. P \equiv \bar{a} \langle P_1 \rangle . P_2, P_1 \in [[A_1]]_{Pr} \text{ and } P_2 \in [[A_2]]_{Pr}\} \\
[[A \setminus \bar{a}]]_{Pr} &= \{P \mid \bar{a} \langle P \rangle . 0 \in [[A]]_{Pr}\} \\
[[A_1 | A_2]]_{Pr} &= \{P \mid \exists Q_1, Q_2. P \equiv Q_1 | Q_2, Q_1 \in [[A_1]]_{Pr} \text{ and } Q_2 \in [[A_2]]_{Pr}\} \\
[[A_1 \triangleright A_2]]_{Pr} &= \{P \mid \forall Q. Q \in [[A_1]]_{Pr} \text{ implies } P | Q \in [[A_2]]_{Pr}\} \\
[[a \textcircled{R} A]]_{Pr} &= \{P \mid \exists Q. P \equiv (\nu a)Q \text{ and } Q \in [[A]]_{Pr}\} \\
[[A \textcircled{o} a]]_{Pr} &= \{P \mid (\nu a)P \in [[A]]_{Pr}\} \\
[[\langle \mathbf{N}x \rangle A]]_{Pr} &= \cup_{n \notin fn((\mathbf{N}x)A)} ([[A\{n/x\}]]_{Pr} \setminus \{P \mid n \in fn(P)\}) \\
[[\langle \mathbf{N}X \rangle A]]_{Pr} &= \cup_{V \notin fpv((\mathbf{N}X)A)} ([[A\{V/X\}]]_{Pr} \setminus \{P \mid V \in fpv(P)\}) \\
[[\langle \ominus a \rangle A]]_{Pr} &= \{P \mid a \notin fn(P) \text{ and } P \in [[A]]_{Pr}\} \\
[[\langle \tilde{\ominus} \rangle A]]_{Pr} &= \{P \mid \exists Q. P \equiv Q \text{ and } bn(Q) = \emptyset \text{ and } Q \in [[A]]_{Pr}\} \\
[[a \neq b]]_{Pr} &= Pr \text{ if } a \neq b \\
[[a \neq b]]_{Pr} &= \emptyset \text{ if } a = b
\end{aligned}$$

In SL , formula $\langle a \langle A_1 \rangle \rangle A_2$ describes processes that can receive a process satisfying A_1 and then becomes a process satisfying A_2 . Formula $\langle a[A_1] \rangle A_2$ describes processes that if it receive any process satisfying A_1 then it becomes a process satisfying A_2 . $A \setminus a \odot X$ is an adjunct operator of $a \odot X.A$, and $A \setminus \bar{a}$ is an adjunct operator of $\bar{a} \langle A \rangle . 0$. $(\ominus a)A$ represents processes that satisfying A and a is not its free name. $(\tilde{\ominus})A$ represents processes that satisfy A and have no bound names. Other operators in SL are well known in spatial logic or can be interpreted similarly as above operators.

Definition 7 $P \models_{SL} A$ iff $P \in [[A]]_{Pr}$.

Definition 8 For a set of formulas Γ and a formula A , we write $\Gamma \models_{SL} A$, if A is valid in all processes in which Γ is satisfiable.

Definition 9 If “ A_1, \dots, A_n infer B ” is an instance of an inference rule, and if the formulas A_1, \dots, A_n have appeared earlier in the proof, then we say that B follows from an application of an inference rule. A proof is said to be from Γ to A if the premise is Γ and the last formula is A in the proof. We say A is provable from Γ in SL , and write $\Gamma \vdash_{SL} A$, if there is a proof from Γ to A in SL .

For example, the following sets can be defined by operators in SL :

$$\{P \mid \forall P_1. P_1 \in [[A_1]]_{Pr} \text{ implies } \bar{a}\langle P_1 \rangle.P \in [[A_2]]_{Pr}\} = [[(b\langle Y \rangle.\bar{a}\langle A_1 \rangle.Y \triangleright \langle \tau \rangle A_2) \setminus \bar{b}]]_{Pr}$$

$$\{P \mid \forall P_1. P_1 \in [[A_1]]_{Pr} \text{ implies } \bar{a}\langle P \rangle.P_1 \in [[A_2]]_{Pr}\} = [[(b\langle Y \rangle.\bar{a}\langle Y \rangle.A_1 \triangleright \langle \tau \rangle A_2) \setminus \bar{b}]]_{Pr}$$

$$\{P \mid a \in fn(P) \text{ and } P \in [[A]]_{Pr}\} = [[\neg(\ominus a)\top \wedge A]]_{Pr}$$

$$\{P \mid X \in fv(P) \text{ and } P \in [[A]]_{Pr}\} = [[\neg(\ominus X)\top \wedge A]]_{Pr}$$

$(\mathbf{H}x)A = (\mathbf{N}x)x\mathbb{R}A$, which is related to name restriction in an appropriate way; namely, that if process P satisfies formulas $A\{n/x\}$, then $(\nu n)P$ satisfies $(\mathbf{H}x)A$.

$(a\mathbf{H}X)A = (\mathbf{N}X)a \odot X.A$, which is related to process variable restriction in an appropriate way; namely, that if process P satisfies formulas $A\{U/X\}$, then $a(U).P$ satisfies $(a\mathbf{H}X)A$.

3.2 Inference System of SL

Now we list a number of valid properties of spatial logic. The combination of the complete inference system of first order logic and the following axioms and rules form the inference system of SL .

$$\begin{array}{lll} \langle \alpha \rangle \perp \rightarrow \perp & \perp \triangleright A \rightarrow \perp & (A|B)|C \leftrightarrow A|(B|C) \\ a \odot X.\perp \rightarrow \perp & a\mathbb{R}\perp \rightarrow \perp & A|0 \leftrightarrow A \\ \bar{a}\langle \top \rangle.\perp \rightarrow \perp & \perp \odot a \rightarrow \perp & a\mathbb{R}0 \leftrightarrow 0 \\ \bar{a}\langle \perp \rangle.\top \rightarrow \perp & (\ominus a)\perp \rightarrow \perp & a\mathbb{R}b\mathbb{R}A \leftrightarrow b\mathbb{R}a\mathbb{R}A \\ \perp \setminus a \odot X \rightarrow \perp & (\mathbf{N}x)\perp \rightarrow \perp & a\mathbb{R}((\ominus a)A|B) \leftrightarrow (\ominus a)A|a\mathbb{R}B \\ \perp \setminus \bar{a} \rightarrow \perp & (\tilde{\ominus})\perp \rightarrow \perp & a\mathbb{R}A \rightarrow (\mathbf{N}b)b\mathbb{R}A\{b/a\} \\ A|\perp \rightarrow \perp & (\mathbf{N}X)\perp \rightarrow \perp & a \odot X.A \rightarrow (\mathbf{N}Y)a \odot Y.A\{Y/X\} \\ A \triangleright \perp \rightarrow \perp & A|B \leftrightarrow B|A & (\ominus a)0 \leftrightarrow 0 \end{array}$$

$$\begin{array}{ll} (\ominus a)X \leftrightarrow X & (\tilde{\ominus})0 \leftrightarrow 0 \\ (\ominus a)a(X).A \leftrightarrow \perp & (\tilde{\ominus})X \leftrightarrow X \\ (\ominus a)\bar{a}\langle B \rangle.A \leftrightarrow \perp & (\tilde{\ominus})a \odot X.A \leftrightarrow a \odot X.(\tilde{\ominus})A \\ a \neq b \rightarrow ((\ominus a)b(X).A \leftrightarrow b(X).(\ominus a)A) & (\tilde{\ominus})\bar{a}\langle B \rangle.A \leftrightarrow \bar{a}\langle (\tilde{\ominus})B \rangle.(\tilde{\ominus})A \\ a \neq b \rightarrow ((\ominus a)\bar{b}\langle B \rangle.A \leftrightarrow \bar{b}\langle (\ominus a)B \rangle.(\ominus a)A) & (\tilde{\ominus})A|(\tilde{\ominus})B \leftrightarrow (\tilde{\ominus})(A|B) \\ (\ominus a)A|(\ominus a)B \leftrightarrow (\ominus a)(A|B) & (\tilde{\ominus})a\mathbb{R}A \rightarrow \perp \\ a \neq b \rightarrow ((\ominus a)(\ominus b)A \leftrightarrow (\ominus b)(\ominus a)A) & \\ (\ominus a)\top \rightarrow (\ominus a)a\mathbb{R}A & \end{array}$$

$$\begin{array}{ll}
(\mathbf{N}x)0 \leftrightarrow 0 & (\mathbf{N}X)0 \leftrightarrow 0 \\
(\mathbf{N}x)X \leftrightarrow X & (\mathbf{N}X)X \rightarrow Y \\
(\mathbf{N}x)a \odot X.A \leftrightarrow a \odot X.(\mathbf{N}x)(x \neq a \wedge A) & (\mathbf{N}X)a \odot Y.A \leftrightarrow a \odot Y.(\mathbf{N}X)A \\
(\mathbf{N}x)\bar{a}\langle B \rangle.A \rightarrow \bar{a}\langle (\mathbf{N}x)(x \neq a \wedge B) \rangle.(\mathbf{N}x)(x \neq a \wedge A) & (\mathbf{N}X)\bar{a}\langle B \rangle.A \rightarrow \bar{a}\langle (\mathbf{N}X)B \rangle.(\mathbf{N}X)A \\
(\mathbf{N}x)(A|B) \rightarrow (\mathbf{N}x)A|(\mathbf{N}x)B & (\mathbf{N}X)(A|B) \rightarrow (\mathbf{N}X)A|(\mathbf{N}X)B \\
(\mathbf{N}x)x \neq a \wedge a\mathbb{R}A \rightarrow a\mathbb{R}(\mathbf{N}x)A & (\mathbf{N}X)a\mathbb{R}A \leftrightarrow a\mathbb{R}(\mathbf{N}X)A
\end{array}$$

$$\begin{array}{lll}
a \odot X.(A \setminus a \odot X) \rightarrow A & A \rightarrow (a\mathbb{R}A \odot a) & A \setminus a \odot X, A \rightarrow B \vdash B \setminus a \odot X \\
A \rightarrow (a \odot X.A) \setminus a \odot X) & \langle \alpha \rangle A, A \rightarrow B \vdash \langle \alpha \rangle B & A \setminus \bar{a}, A \rightarrow B \vdash B \setminus \bar{a} \\
\bar{a}\langle A \setminus \bar{a} \rangle.0 \rightarrow A & a \odot X.A, A \rightarrow B \vdash a \odot X.B & A \rightarrow B \vdash A|C \rightarrow B|C \\
A \rightarrow ((\bar{a}\langle A \rangle.0) \setminus \bar{a}) & \bar{a}\langle C \rangle.A, A \rightarrow B \vdash \bar{a}\langle C \rangle.B & a\mathbb{R}A, A \rightarrow B \vdash a\mathbb{R}B \\
(A|A \triangleright B) \rightarrow B & \bar{a}\langle B \rangle.A, B \rightarrow C \vdash \bar{a}\langle C \rangle.A & (\ominus a)A, A \rightarrow B \vdash (\ominus a)B \\
A \rightarrow (B \triangleright A|B) & \langle \bar{a}\langle B \rangle \rangle A, C \rightarrow B \vdash \langle \bar{a}\langle C \rangle \rangle A & (\tilde{\ominus})A, A \rightarrow B \vdash (\tilde{\ominus})B \\
a\mathbb{R}(A \odot a) \rightarrow A & \langle a[B] \rangle A, C \rightarrow B \vdash \langle a[C] \rangle A &
\end{array}$$

$$\begin{array}{l}
\bar{a}\langle B \rangle.A \rightarrow \langle \bar{a}\langle B \rangle \rangle A \\
(a \odot U.A \wedge ((\tilde{\ominus})B \leftrightarrow B)) \rightarrow \langle a[B] \rangle A\{B/U\} \\
(\langle \tau \rangle A)|B \rightarrow \langle \tau \rangle (A|B) \\
(\langle a\langle C \rangle \rangle A)|B \rightarrow \langle a\langle C \rangle \rangle (A|B) \\
(((\ominus b_1, \dots, \ominus b_n)B \leftrightarrow B) \wedge ((\tilde{\ominus})C \leftrightarrow C)) \rightarrow \\
\quad ((\langle \bar{a}\langle b_1\mathbb{R}\dots b_n\mathbb{R}C \rangle \rangle A)|B \rightarrow \langle \bar{a}\langle b_1\mathbb{R}\dots b_n\mathbb{R}C \rangle \rangle (A|B)) \\
(((\ominus b_1, \dots, \ominus b_n)B \leftrightarrow B) \wedge ((\tilde{\ominus})C \leftrightarrow C)) \rightarrow \\
\quad ((\langle \bar{a}\langle b_1\mathbb{R}\dots b_n\mathbb{R}C \rangle \rangle A)|\langle a[C] \rangle B \rightarrow \langle \tau \rangle b_1\mathbb{R}\dots b_n\mathbb{R}(A|B)) \\
(a \neq b \wedge ((\ominus a)B \leftrightarrow B) \wedge ((\tilde{\ominus})B \leftrightarrow B)) \rightarrow (a\mathbb{R}\langle b\langle B \rangle \rangle A \rightarrow \langle b\langle B \rangle \rangle a\mathbb{R}A) \\
(\wedge_{i=1}^n a \neq b_i \wedge a \neq c \wedge ((\ominus a)B \leftrightarrow B) \wedge ((\tilde{\ominus})B \leftrightarrow B)) \rightarrow \\
\quad (a\mathbb{R}\langle \bar{c}\langle b_1\mathbb{R}\dots b_n\mathbb{R}B \rangle \rangle A \rightarrow \langle \bar{c}\langle b_1\mathbb{R}\dots b_n\mathbb{R}B \rangle \rangle a\mathbb{R}A) \\
(a \neq b \wedge \wedge_{i=1}^n b \neq c_i \wedge (B \rightarrow \neg(\ominus b)\top) \wedge ((\tilde{\ominus})B \leftrightarrow B)) \rightarrow \\
\quad (b\mathbb{R}\langle \bar{a}\langle c_1\mathbb{R}\dots c_n\mathbb{R}B \rangle \rangle A \rightarrow \langle \bar{a}\langle b\mathbb{R}c_1\mathbb{R}\dots c_n\mathbb{R}B \rangle \rangle A) \\
\langle a[B] \rangle A \rightarrow \langle a\langle B \rangle \rangle A \\
\langle a\langle B \rangle \rangle A \rightarrow \langle a[B] \rangle A, \text{ where } B \text{ is syntactically a valid process in the higher} \\
\text{order } \pi\text{-calculus.}
\end{array}$$

Intuitively, axiom $a\mathbb{R}A \rightarrow (\mathbf{N}b)b\mathbb{R}A\{b/a\}$ means that if process P satisfies $(\nu a)A$ and b is a fresh name then P satisfies $(\nu b)A\{b/a\}$. Axiom $\bar{a}\langle B \rangle.A \rightarrow \langle \bar{a}\langle B \rangle \rangle A$ means that an output prefix process can perform an output action, which is a spatial logical version of Rule *OUT* in the labelled transition system of higher order π -calculus. Axiom $(a \odot U.A \wedge ((\tilde{\ominus})B \leftrightarrow B)) \rightarrow \langle a[B] \rangle A\{B/U\}$ means that an input prefix process can perform an input action, which is a spatial logical version of Rule *IN* in the labelled transition system of higher order π -calculus. Axiom $((\ominus b_1, \dots, \ominus b_n)B \leftrightarrow B) \wedge ((\tilde{\ominus})C \leftrightarrow C) \rightarrow ((\langle \bar{a}\langle b_1\mathbb{R}\dots b_n\mathbb{R}C \rangle \rangle A)|\langle a[C] \rangle B \rightarrow \langle \tau \rangle b_1\mathbb{R}\dots b_n\mathbb{R}(A|B))$ is a spatial logical version of Rule *COM*. Other axioms and rules are spatial logical version of structure congruence rules or labelled transition rules similarly.

3.3 Soundness of SL

Inference system of SL is said to be sound with respect to processes if every formula provable in SL is valid with respect to processes.

Now, we can prove the soundness of inference system of SL :

Proposition 1 $\Gamma \vdash_{SL} A \Rightarrow \Gamma \models_{SL} A$

Proof. See Appendix A.

3.4 Incompleteness of SL

The system SL is complete with respect to processes if every formula valid with respect to processes is provable in SL . For a logic, completeness is an important property. The soundness and completeness provide a tight connection between the syntactic notion of provability and the semantic notion of validity. Unfortunately, by the compactness property [18], inference system of SL is not complete.

The depth of higher order processes in Pr , is defined as below:

Definition 10 $d(0) = 0$; $d(U) = 0$; $d(a(U).P) = 1 + d(P)$; $d(\bar{a}\langle E \rangle.P) = 1 + d(E) + d(P)$; $d(P_1|P_2) = d(P_1) + d(P_2)$; $d((\nu a)P) = d(P)$.

Lemma 1 For any $P \in Pr$, there exists n , such that $d(P) = n$.

Proof. Induction on the structure of P .

Proposition 2 There is no finite inference system such that $\Gamma \models_{SL} A \Rightarrow \Gamma \vdash_{SL} A$.

Proof. See Appendix B.

3.5 Spatial Logic as a Specification of Processes

In the refinement calculus [23], imperative programming languages are extended by specification statements, which specify parts of a program “yet to be developed”. Then the development of a program begins with a specification statement, and ends with a executable program by refining a specification to its possible implementations. In this paper, we generalize this idea to the case of process calculi. Roughly speaking, we extend processes to spatial logic formulas which are regarded as the specification statements. Processes can be regarded as a special kind of spatial logic. One can view the intensional operators of spatial logic as the “executable program statements”, for example, $\bar{a}\langle P \rangle.Q$, $P|Q$ and etc; and view the extensional operators of spatial logic as the “specification statements”, for example, $A \triangleright B$, $A \setminus \bar{b}$ and etc. For example, $(b \odot Y.\bar{a}\langle Y \rangle.A_1 \triangleright \langle \tau \rangle A_2) \setminus \bar{b} | (d \odot Y.\bar{c}\langle B_1 \rangle.Y \triangleright \langle \tau \rangle B_2) \setminus \bar{d}$ represents a specification statement which describes a process consisting of a parallel of two processes satisfying statements $(b \odot Y.\bar{a}\langle Y \rangle.A_1 \triangleright \langle \tau \rangle A_2) \setminus \bar{b}$ and $(d \odot Y.\bar{c}\langle B_1 \rangle.Y \triangleright \langle \tau \rangle B_2) \setminus \bar{d}$ respectively. Furthermore, $(b \odot Y.\bar{a}\langle Y \rangle.A_1 \triangleright \langle \tau \rangle A_2) \setminus \bar{b}$ represents a specification which describes a process P that $\bar{a}\langle P \rangle.Q$ satisfying A_2 for any Q satisfying A_1 . Similarly, $(d \odot Y.\bar{c}\langle B_1 \rangle.Y \triangleright \langle \tau \rangle B_2) \setminus \bar{d}$ represents a specification statement which describes a process M such that $\bar{c}\langle N \rangle.M$ satisfying B_2 for any N satisfying B_1 . We can also define refinement relation on spatial logic formulas. Intuitively, if

$\models_{SL} A \rightarrow B$, then A refines B . For example, $a\textcircled{R}(a \odot X.d.X|\bar{a}\langle c.0\rangle.e.0)$ refines $a\textcircled{R}(\langle a[c.0]\rangle d.c.0|\langle \bar{a}\langle c.0\rangle\rangle e.0)$. Based on spatial logic, one may develop a theory of refinement for concurrent processes. This will be a future research direction for us.

3.6 Processes as Special Formulas of Spatial Logic

Any process can be regarded as a special formulas of spatial logic. For example, $(\mathbf{N}a)a\textcircled{R}(\mathbf{N}X)(a \odot X.d.X|\bar{a}\langle c.0\rangle.e.0)$ is a spatial logic formula, which represents the process which is structural congruent to $(\nu a)(a(X).d.X|\bar{a}\langle c.0\rangle.e.0)$. Furthermore, in this section, we will show that structural congruence and labelled transition relation can be reformulated as the logical relation of spatial logical formulas.

Definition 11 The translating function T^{PS} is defined inductively as follows:

$T^{PS}(P) \stackrel{def}{=} P$ for process P that has no operators of $(\nu a)\cdot$, or $a(X)\cdot$;

$T^{PS}((\nu a)P) \stackrel{def}{=} (\mathbf{H}a)T^{PS}(P)$;

$T^{PS}(a(X).P) \stackrel{def}{=} (a\mathbf{H}X)T^{PS}(P)$.

Proposition 3 For any $P, Q \in Pr^c$, $P \equiv Q \Leftrightarrow P \models_{SL} T^{PS}(Q)$ and $Q \models_{SL} T^{PS}(P) \Leftrightarrow T^{PS}(P) \vdash_{SL} T^{PS}(Q)$ and $T^{PS}(Q) \vdash_{SL} T^{PS}(P)$.

Proof. See Appendix C.

Proposition 4 For any $P, Q \in Pr^c$, $P \xrightarrow{\alpha} Q \Leftrightarrow P \models_{SL} \langle \alpha \rangle T^{PS}(Q) \Leftrightarrow T^{PS}(P) \vdash_{SL} \langle \alpha \rangle T^{PS}(Q)$.

Proof. See Appendix D.

Although Proposition 2 states that the inference system is not completeness, Propositions 3 and 4 show that this inference system is completeness with respect to structural congruence and labelled transition relation of processes.

3.7 Behavioral Equivalence Relation of Spatial Logic

In [9], we introduced a spatial logic called L , and proved that L gives a characterization of context bisimulation.

Definition 12 [9] Syntax of logic L

$A ::= \neg A \mid A_1 \wedge A_2 \mid \langle a\langle \top \rangle \rangle \top \mid \langle \bar{a}\langle \top \rangle \rangle \top \mid \langle \tau \rangle A \mid A_1 \triangleright A_2$.

It is easy to see that L is a sublogic of SL .

In [9], we proved the equivalence between \sim_{Ct} and logical equivalence with respect to L .

Proposition 5 [9] For any $P, Q \in Pr^c$, $P \sim_{Ct} Q \Leftrightarrow$ for any formula $A \in L$, $P \models_L A$ iff $Q \models_L A$.

Definition 13 A and B are behavioral equivalent with respect to L , written $A \sim_L B$, iff for any formula $C \in L$, $\models_{SL} A \rightarrow C$ iff $\models_{SL} B \rightarrow C$.

By Proposition 5, it is easy to get the following corollary, which characterize \sim_{Ct} by SL property.

Corollary 1 For any $P, Q \in Pr^c$, $P \sim_{Ct} Q \Leftrightarrow P \sim_L Q$.

Relation \sim_L is a binary relation on spatial logical formulas. The above results show that \sim_L gives a logical characterization of bisimulation when formulas

are in the form of processes. Moreover, relation \sim_L also gives a possibility to generalize bisimulation on processes to that on spatial logical formulas. Since we have discussed that spatial logical formulas can be regarded as specifications of processes, we may get a concept of bisimulation on specifications of processes based on \sim_L .

4 Logics for Weak Semantics

In this section, we present a logic for weak semantics, named WL . Roughly speaking, in this logic, action temporal operators $\langle\tau\rangle$, $\langle a\langle A\rangle\rangle$, $\langle a[A]\rangle$ and $\langle\bar{a}\langle A\rangle\rangle$ in SL are replaced by the weak semantics version of operators $\langle\langle\varepsilon\rangle\rangle$, $\langle\langle a\langle A\rangle\rangle\rangle$, $\langle\langle a[A]\rangle\rangle$ and $\langle\langle\bar{a}\langle A\rangle\rangle\rangle$. Almost all definitions and results of SL can be generalized to WL .

4.1 Syntax and Semantics of Logic WL

Now we introduce a logic called WL , which is a weak semantics version of spatial logic.

Definition 14 Syntax of logic WL

$$A ::= \top \mid \perp \mid \neg A \mid A_1 \wedge A_2 \mid \langle\langle\varepsilon\rangle\rangle A \mid \langle\langle a\langle A_1\rangle\rangle\rangle A_2 \mid \langle\langle a[A_1]\rangle\rangle A_2 \mid \langle\langle\bar{a}\langle A_1\rangle\rangle\rangle A_2 \mid 0 \mid X \mid a \odot X.A \mid A \setminus a \odot X \mid \bar{a}\langle A_1\rangle.A_2 \mid A \setminus \bar{a} \mid A_1 \mid A_2 \mid A_1 \triangleright A_2 \mid a \textcircled{R} A \mid A \odot a \mid (\mathbf{N}x)A \mid (\mathbf{N}X)A \mid (\ominus a)A \mid (\tilde{\ominus})A \mid a \neq b$$

Definition 15 Semantics of logic WL

Semantics of formulas of WL can be the same as formulas of SL , except that semantics of operators $\langle\langle\varepsilon\rangle\rangle$, $\langle\langle a\langle A\rangle\rangle\rangle$, $\langle\langle a[A]\rangle\rangle$ and $\langle\langle\bar{a}\langle A\rangle\rangle\rangle$ should be defined as follows:

$$\begin{aligned} [[\langle\langle\varepsilon\rangle\rangle A]]_{Pr} &= \{P \mid \exists Q. P \xrightarrow{\varepsilon} Q \text{ and } Q \in [[A]]_{Pr}\} \\ [[\langle\langle a\langle A_1\rangle\rangle\rangle A_2]]_{Pr} &= \{P \mid \exists P_1, P_2. P \xrightarrow{a(P_1)} P_2, P_1 \in [[A_1]]_{Pr} \text{ and } P_2 \in [[A_2]]_{Pr}\} \\ [[\langle\langle a[A_1]\rangle\rangle A_2]]_{Pr} &= \{P \mid \forall R, R \in [[A_1]]_{Pr}, \exists Q. P \xrightarrow{a(R)} Q \text{ and } Q \in [[A_2]]_{Pr}\} \\ [[\langle\langle\bar{a}\langle A_1\rangle\rangle\rangle A_2]]_{Pr} &= \{P \mid \exists P_1, P_2. P \xrightarrow{(\nu\tilde{b})\bar{a}(P_1)} P_2, (\nu\tilde{b})P_1 \in [[A_1]]_{Pr} \text{ and } P_2 \in [[A_2]]_{Pr}\} \end{aligned}$$

4.2 Inference System of WL

The inference system of WL is similar to the inference system of SL except that any inference rule about action temporal operators $\langle\tau\rangle$, $\langle a\langle A\rangle\rangle$, $\langle a[A]\rangle$ and $\langle\bar{a}\langle A\rangle\rangle$ in SL is replaced by one of the following inference rules.

$$\begin{aligned} &\langle\langle\alpha\rangle\rangle\perp \rightarrow \perp \\ &\langle\langle\alpha\rangle\rangle A, A \rightarrow B \vdash \langle\langle\alpha\rangle\rangle B \\ &\langle\langle\alpha\rangle\rangle A, A \rightarrow \langle\langle\varepsilon\rangle\rangle B \vdash \langle\langle\alpha\rangle\rangle B \\ &\langle\langle\varepsilon\rangle\rangle A, A \rightarrow \langle\langle\alpha\rangle\rangle B \vdash \langle\langle\alpha\rangle\rangle B \\ &\langle\langle\bar{a}\langle B\rangle\rangle\rangle A, C \rightarrow B \vdash \langle\langle\bar{a}\langle C\rangle\rangle\rangle A \\ &\langle\langle a[B]\rangle\rangle A, C \rightarrow B \vdash \langle\langle a[C]\rangle\rangle A \end{aligned}$$

$$\begin{aligned}
& \bar{a}\langle B \rangle.A \rightarrow \langle \langle \bar{a}\langle B \rangle \rangle \rangle A \\
& (a \odot U.A \wedge ((\tilde{\ominus})B \leftrightarrow B)) \rightarrow \langle \langle a[B] \rangle \rangle A\{B/U\} \\
& (\langle \langle \varepsilon \rangle \rangle A)|B \rightarrow \langle \langle \varepsilon \rangle \rangle (A|B) \\
& (\langle \langle a\langle C \rangle \rangle \rangle A)|B \rightarrow \langle \langle a\langle C \rangle \rangle \rangle (A|B) \\
& (((\ominus b_1, \dots, \ominus b_n)B \leftrightarrow B) \wedge ((\tilde{\ominus})C \leftrightarrow C)) \rightarrow \\
& \quad (((\langle \langle \bar{a}\langle b_1 \mathbb{R} \dots b_n \mathbb{R} C \rangle \rangle \rangle A)|B \rightarrow \langle \langle \bar{a}\langle b_1 \mathbb{R} \dots b_n \mathbb{R} C \rangle \rangle \rangle (A|B)) \\
& (((\ominus b_1, \dots, \ominus b_n)B \leftrightarrow B) \wedge ((\tilde{\ominus})C \leftrightarrow C)) \rightarrow \\
& \quad (((\langle \langle \bar{a}\langle b_1 \mathbb{R} \dots b_n \mathbb{R} C \rangle \rangle \rangle A)|\langle \langle a[C] \rangle \rangle B \rightarrow \langle \langle \varepsilon \rangle \rangle b_1 \mathbb{R} \dots b_n \mathbb{R} (A|B)) \\
& a\mathbb{R}\langle \langle \varepsilon \rangle \rangle A \rightarrow \langle \langle \varepsilon \rangle \rangle a\mathbb{R}A \\
& (a \neq b \wedge (((\ominus a)B \wedge (\tilde{\ominus})B) \leftrightarrow B)) \rightarrow (a\mathbb{R}\langle \langle b\langle B \rangle \rangle \rangle A \rightarrow \langle \langle b\langle B \rangle \rangle \rangle a\mathbb{R}A) \\
& (\wedge_{i=1}^n a \neq b_i \wedge a \neq c \wedge ((\ominus a)B \leftrightarrow B) \wedge ((\tilde{\ominus})B \leftrightarrow B)) \rightarrow \\
& \quad (a\mathbb{R}\langle \langle \bar{c}\langle b_1 \mathbb{R} \dots b_n \mathbb{R} B \rangle \rangle \rangle A \rightarrow \langle \langle \bar{c}\langle b_1 \mathbb{R} \dots b_n \mathbb{R} B \rangle \rangle \rangle a\mathbb{R}A) \\
& (a \neq b \wedge \wedge_{i=1}^n b \neq c_i \wedge (B \rightarrow \neg(\ominus b)\top) \wedge ((\tilde{\ominus})B \leftrightarrow B)) \rightarrow \\
& \quad (b\mathbb{R}\langle \langle \bar{a}\langle c_1 \mathbb{R} \dots c_n \mathbb{R} B \rangle \rangle \rangle A \rightarrow \langle \langle \bar{a}\langle b\mathbb{R}c_1 \mathbb{R} \dots c_n \mathbb{R} B \rangle \rangle \rangle A) \\
& \langle \langle a[B] \rangle \rangle A \rightarrow \langle \langle a\langle B \rangle \rangle \rangle A \\
& \langle \langle a\langle B \rangle \rangle \rangle A \rightarrow \langle \langle a[B] \rangle \rangle A, \text{ where } B \text{ is syntactically a valid process in the} \\
& \text{higher order } \pi\text{-calculus.}
\end{aligned}$$

The above axioms and rules are weak semantics version of corresponding axioms and rules in SL .

The soundness and incompleteness of inference system of WL can be given similarly as the case of SL :

Proposition 6 $\Gamma \vdash_{WL} A \Rightarrow \Gamma \models_{WL} A$

Proposition 7 There is no finite inference system such that $\Gamma \models_{WL} A \Rightarrow \Gamma \vdash_{WL} A$.

Similar to Proposition 4, we show that many-steps transition relation is provable in WL .

Proposition 8 For any $P, Q \in Pr^c$, $P \xRightarrow{\alpha} Q \Leftrightarrow P \models_{WL} \langle \langle \alpha \rangle \rangle T^{PS}(Q) \Leftrightarrow T^{PS}(P) \vdash_{WL} \langle \langle \alpha \rangle \rangle T^{PS}(Q)$.

Since structural congruence and labelled transition relation are central concepts in the theory of processes, and they can be characterized in WL , the above propositions give a possible approach to reduce the theory of processes to the theory of spatial logic in the case of weak semantics.

5 Adding μ -Operator to SL

In this section, we add μ -operator [3] to SL . We call this new logic as μSL . We will show that WL is a sublogic of μSL .

5.1 Syntax and Semantics of μSL

The formula of μSL is the same as the formula of SL except that the following μ -calculus formula is added:

If $A(X) \in \mu SL$, then $\mu X.A(X) \in \mu SL$, here X occurs positively in $A(X)$, i.e., all free occurrences of X fall under an even number of negations..

The model of μSL is the same as SL . We write such set of processes in which A is true as $[[A]]_{Pr}^e$, where $e: Var \rightarrow 2^{Pr}$ is an environment. We denote by $e[X \leftarrow W]$ a new environment that is the same as e except that $e[X \leftarrow W](X) = W$. The set $[[A]]_S^e$ is the set of processes that satisfy A . In the following, we abbreviate $A(B)$ as $A\{B/X\}$, and abbreviate $A^{n+1}(B)$ as $A(A^n(B))$ where $A^0(B)$ is B .

Semantics of μ -operator is given as following:

$$[[\mu X.A(X)]]_{Pr}^e = \cap \{W \subseteq Pr \mid [[A(X)]]_{Pr}^{e[X \leftarrow W]} \subseteq W\}.$$

In μ -calculus [3], it is well known that $[[\mu X.A(X)]]_{Pr}^e = [[A^1(\perp)]]_{Pr}^e \cup [[A^2(\perp)]]_{Pr}^e \cup \dots$

5.2 Inference System of μSL

Inference system of μSL is the combination of the following two rules of μ -calculus [3] and the inference system of SL .

$$A(\mu X.A(X)) \rightarrow \mu X.A(X).$$

$$\vdash A(B) \rightarrow B \Rightarrow \vdash \mu X.A(X) \rightarrow B.$$

The soundness and incompleteness of inference system of μSL can be given as the case of SL .

Proposition 9 $\Gamma \vdash_{\mu SL} A \Rightarrow \Gamma \models_{\mu SL} A$

Proposition 10 There is no finite inference system such that $\Gamma \models_{\mu SL} A \Rightarrow \Gamma \vdash_{\mu SL} A$.

5.3 Expressivity of μSL

In this section, we will discuss the express power of μSL . We will prove that WL is a sublogic of μSL and give a function which can translates a WL formula into an equivalent μSL formula.

Now we can give a translating function from WL formula to μSL formula:

Definition 16 The translating function T is defined inductively as follows:

$$T^{WM}(A) \stackrel{def}{=} A \text{ for proposition } A \text{ of } WL \text{ that is not in the form of } \langle\langle\varepsilon\rangle\rangle A, \langle\langle a(A_1) \rangle\rangle A_2, \langle\langle a[A_1] \rangle\rangle A_2 \text{ or } \langle\langle \bar{a}(A_1) \rangle\rangle A_2.$$

$$T^{WM}(\langle\langle\varepsilon\rangle\rangle A) \stackrel{def}{=} \mu X.(T^{WM}(A) \vee \langle\tau\rangle X)$$

$$T^{WM}(\langle\langle a(A_1) \rangle\rangle A_2) \stackrel{def}{=} \mu X.(\langle a(T^{WM}(A_1)) \rangle (\mu Y.(T^{WM}(A_2) \vee \langle\tau\rangle Y)) \vee \langle\tau\rangle X)$$

$$T^{WM}(\langle\langle a[A_1] \rangle\rangle A_2) \stackrel{def}{=} \mu X.(\langle a[T^{WM}(A_1)] \rangle (\mu Y.(T^{WM}(A_2) \vee \langle\tau\rangle Y)) \vee \langle\tau\rangle X)$$

$$T^{WM}(\langle\langle \bar{a}(A_1) \rangle\rangle A_2) \stackrel{def}{=} \mu X.(\langle \bar{a}(T^{WM}(A_1)) \rangle (\mu Y.(T^{WM}(A_2) \vee \langle\tau\rangle Y)) \vee \langle\tau\rangle X)$$

The following proposition states the correctness of translating function T^{WM} .

Proposition 11 For any $A \in WL$, $T^{WM}(A) \in \mu SL$; for any $P \in Pr$, $P \models_{\mu SL} T^{WM}(A) \Leftrightarrow P \models_{WL} A$.

Proof : See Appendix E.

In μSL , we can also define the replication operator:

$$\textbf{Definition 17 } !A \stackrel{def}{=} \neg \mu X. \neg (A \mid \neg X)$$

Proposition 12 $\vdash_{\mu SL} A \mid !A \leftrightarrow !A$

Proof : See Appendix F.

The above results show that WL is a sublogic of μSL . Therefore μSL can be used as a uniform logic framework to study both the strong semantics and the weak semantics of higher order π -calculus.

6 Conclusions

Spatial logic was proposed to describe structural and behavioral properties of processes. There are many papers on spatial logic and process calculi. Spatial logic is related to some topics on process calculi, such as model checking, structural congruence, bisimulation and type system. In [16], a spatial logic for ambients calculus was studied, and a model checking algorithm was proposed. Some axioms of spatial logic were given, but the soundness and completeness of logic was not studied. Most spatial logics for concurrency are intensional [27], in the sense that they induce an equivalence that coincides with structural congruence, which is much finer than bisimilarity. In [22], Hirschhoff studied an extensional spatial logic. This logic only has spatial composition adjunct (\triangleright), revelation adjunct (\oslash), a simple temporal modality ($\langle \rangle$), and an operator for fresh name quantification. For π -calculus, this extensional spatial logic was proven to induce the same separative power as strong early bisimilarity. In [9], context bisimulation of higher order π -calculus was characterized by an extensional spatial logic. In [5], a type system of processes based on spatial logic was given, where types are interpreted as formulas of spatial logic.

In this paper, we want to show that the theory of processes can be reduced to the theory of spatial logics. We firstly defined a logic SL , which comprises some temporal operators and spatial operators. We gave the inference system of SL and showed the soundness and incompleteness of SL . Furthermore, we showed that structural congruence and transition relation of higher order π -calculus can be reduced to the logical relation of SL formulas. We also showed that bisimulations in higher order π -calculus can be characterized by a sublogic of SL . Furthermore, we propose a weak semantics version of SL , called WL . At last, we add μ -operator to SL . The new logic named μSL . the expressive power of μSL is studied. These results can be generalized to other process calculi. Since some important concepts of processes can be described in spatial logic, we think that this paper may give an approach of reducing the study of processes to the study of spatial logic. The further work for us is to develop a refinement calculus [23] for concurrent processes based on our spatial logic.

Acknowledgment

This work was supported by the Aviation Science Fund of China under Grant No. 20085552023, the National Natural Science Foundation of China under Grants No. 60473036, No. 60873025, the Natural Science Foundation of Jiangsu Province of China under Grant No. BK2008389, and the Foundation of Provincial Key Laboratory for Computer Information Processing Technology of Soochow University under Grant No. KJS0920.

References

1. R. M. Amadio and M. Dam. Reasoning about Higher-order Processes. In TAPSOFT95, LNCS 915, 202-216. 1995.
2. R. M. Amadio. On the Reduction of CHOCS-Bisimulation to π -calculus Bisimulation. In CONCUR93, LNCS 715, 112-126. 1993.
3. A. Arnold and D. Niwinski. Rudiments of μ -calculus. Studies in Logic, Vol 146, North-Holland, 2001.
4. M. Baldamus and J. Dingel. Modal Characterization of Weak Bisimulation for Higher-order Processes. In TAPSOFT97, LNCS 1214, 285-296, 1997.
5. L. Caires. Spatial-Behavioral Types for Concurrency and Resource Control in Distributed Systems. In Theoretical Computer Science 402(2-3), 2008.
6. L. Caires. Logical Semantics of Types for Concurrency . In CALCO'07, LNCS, 2007.
7. L. Caires, H. T. Vieira. Extensionality of Spatial Observations in Distributed Systems. In EXPRESS'2006, ENTCS, 2006.
8. L. Caires. Behavioral and spatial observations in a logic for the π -calculus. In FOSSACS04, LNCS 2987, 72-87, 2004.
9. Z. Cao, A Spatial Logical Characterisation of Context Bisimulation. In Proceeding of ASIAN 2006, LNCS 4435, 232-240, 2006.
10. Z. Cao. More on bisimulations for higher-order π -calculus. In FOSSACS06, LNCS 3921, 63-78, 2006.
11. L. Caires and L. Cardelli. A Spatial Logic for Concurrency (Part II), Theoretical Computer Science, Vol 322(3), 517-565. 2004.
12. L. Caires and L. Cardelli. A Spatial Logic for Concurrency (Part I). Information and Computation, Vol 186(2), 194-235. 2003.
13. W. Charatonik, S. Dal Zilio, A. D. Gordon, S. Mukhopadhyay, and J.-M. Talbot. The complexity of model checking mobile ambients. In FoSSaCS'01, LNCS 2030, 152-167, 2001.
14. W. Charatonik, S. Dal Zilio, A. D. Gordon, S. Mukhopadhyay, J.-M. Talbot. Model Checking Mobile Ambients.
15. L. Cardelli and A. Gordon. Logical Properties of Name Restriction. In Proc. of TLCA'01, LNCS 2044. 2001.
16. L. Cardelli and A. Gordon. Anytime, Anywhere, Modal Logics for Mobile Ambients. In Proc. of POPL'00, pages 365-377. ACM Press, 2000.
17. G. Conforti and G. Ghelli. Decidability of Freshness ,Undecidability of Revelation. In : Proc. of FoSSaCS'04 , LNCS 2987. 2004.
18. C. C. Chang. Model Theory. North-Holland, 1977.
19. L. Caires and E. Lozes. Elimination of Quantifiers and Undecidability in Spatial Logics for Concurrency. In Theoretical Computer Science Volume 358 , Issue 2 (August 2006) Pages: 293 - 314.
20. A. Jeffrey, J. Rathke. Contextual equivalence for higher-order π -calculus revisited. In Proceedings of Mathematical Foundations of Programming Semantics, Elsevier, 2003.
21. D. Hirschhoff, E. Lozes, and D. Sangiorgi. Separability, Expressiveness and Decidability in the Ambient Logic. In Proc. of LICS'02, pages 423-432. IEEE Computer Society, 2002.
22. D. Hirschhoff. An Extensional Spatial Logic for Mobile Processes. CONCUR'04, LNCS 3170, 325-339, 2004, Springer-Verlag.

23. C. Morgan, P. Gardiner, K. Robison, and T. Vickers. On the Refinement Calculus. Springer-Verlag, 1994.
24. R. Milner, J. Parrow, and D. Walker. Modal logics for mobile processes. Theoretical Computer Science, 114(1):149-171, 1993.
25. L. Gregory Meredith, Matthias Radestock: Namespace Logic: A Logic for a Reflective Higher-Order Calculus. TGC 2005: 353-369.
26. J. Parrow. An introduction to the π -calculus. In J. Bergstra, A. Ponse and S. Smolka editors, Handbook of Process Algebra, North-Holland, Amsterdam, 2001.
27. D. Sangiorgi. Extensionality and Intensionality of the Ambient Logic. In Proc. of the 28th POPL, Pages 4-17. ACM Press, 2001.
28. D. Sangiorgi. Bisimulation in higher-order calculi. Information and Computation, 131(2), 1996.
29. D. Sangiorgi. Expressing mobility in process algebras: first-order and higher-order paradigms. Ph.D thesis, Department of Computer Science, University of Edinburgh, 1992.
30. C. Stirling. Modal Logics for Communicating Systems. Theoretical Computer Science, (49):311-347, 1987.
31. B. Thomsen, Plain CHOCS: A second generation calculus for higher order processes. Acta Informatica, Vol 30, 1-59, 1993.

Appendix A. Proof of Proposition 1

Proposition 1 $\Gamma \vdash_{SL} A \Rightarrow \Gamma \models_{SL} A$

Proof. It is enough by proving that every axiom and every inference rule of inference system is sound. We only discuss the following cases:

Case (1): Axiom $a\mathbb{R}((\ominus a)A|B) \leftrightarrow (\ominus a)A|a\mathbb{R}B$.

Suppose $P \in [[a\mathbb{R}((\ominus a)A|B)]]$, then $P \equiv (\nu a)(P_1|P_2)$, $a \notin fn(P_1)$, $P_1 \in [[A]]$ and $P_2 \in [[B]]$. Therefore we have $P \equiv (\nu a)(P_1|P_2) \equiv P_1|(\nu a)P_2$, $P \in [[(\ominus a)A|a\mathbb{R}B]]$. Hence $a\mathbb{R}((\ominus a)A|B) \leftrightarrow (\ominus a)A|a\mathbb{R}B$. The inverse case is similar.

Case (2): Axiom $a \neq b \rightarrow ((\ominus a)\bar{b}\langle B \rangle.A \leftrightarrow \bar{b}\langle (\ominus a)B \rangle.(\ominus a)A)$.

Suppose $a \neq b$ and $P \in [[(\ominus a)\bar{b}\langle B \rangle.A]]$, then $P \equiv \bar{b}\langle P_1 \rangle.P_2$, $a \notin fn(P_1)$, $a \notin fn(P_2)$, $P_1 \in [[B]]$ and $P_2 \in [[A]]$. Therefore we have $P_1 \in [[(\ominus a)B]]$ and $P_2 \in [[(\ominus a)A]]$, $P \in [[\bar{b}\langle (\ominus a)B \rangle.(\ominus a)A]]$. Hence $a \neq b \rightarrow ((\ominus a)\bar{b}\langle B \rangle.A \rightarrow \bar{b}\langle (\ominus a)B \rangle.(\ominus a)A)$. The inverse case is similar.

Case (3): Axiom $(A|A \triangleright B) \rightarrow B$.

Suppose $P \in [[A|A \triangleright B]]$, then $P \equiv P_1|P_2$, $P_1 \in [[A]]$ and $P_2 \in [[A \triangleright B]]$. Therefore, $P \equiv P_1|P_2 \in [[A|A \triangleright B]]$. Hence $(A|A \triangleright B) \rightarrow B$.

Case (4): Axiom $A \rightarrow (B \triangleright A|B)$.

Suppose $P \in [[A]]$, then for any $Q \in [[B]]$, $P|Q \in [[A|B]]$. Hence $A \rightarrow (B \triangleright A|B)$.

Case (5): Axiom $((\ominus b_1, \dots, \ominus b_n)B \leftrightarrow B) \wedge ((\tilde{\ominus})C \leftrightarrow C) \rightarrow ((\langle \bar{a}\langle b_1\mathbb{R}\dots b_n\mathbb{R}C \rangle \rangle A)|B \rightarrow \langle \bar{a}\langle b_1\mathbb{R}\dots b_n\mathbb{R}C \rangle \rangle (A|B))$.

Suppose $P \in [[(\langle \bar{a}\langle b_1\mathbb{R}\dots b_n\mathbb{R}C \rangle \rangle A)|B]]$, then $P \equiv P_1|P_2$, $P_1 \xrightarrow{(\nu b_1, \dots, b_n)\bar{a}\langle Q \rangle} P'_1$, $P'_1 \in [[A]]$, $P_2 \in [[B]]$ and $Q \in [[C]]$. Since $(\ominus b_1, \dots, \ominus b_n)B \leftrightarrow B$, $\{b_1, \dots, b_n\} \cap fn(P_2) = \emptyset$. Therefore we have $P_1|P_2 \xrightarrow{(\nu b_1, \dots, b_n)\bar{a}\langle Q \rangle} P'_1|P_2$. Hence $((\ominus b_1, \dots, \ominus b_n)B$

$\leftrightarrow B) \wedge ((\tilde{\ominus})C \leftrightarrow C)) \rightarrow ((\langle \bar{a}\langle b_1 \mathbb{R} \dots b_n \mathbb{R} C \rangle \rangle A) | B \rightarrow \langle \bar{a}\langle b_1 \mathbb{R} \dots b_n \mathbb{R} C \rangle \rangle (A | B))$.

Case (6): Axiom $((\ominus b_1, \dots, \ominus b_n)B \leftrightarrow B) \wedge ((\tilde{\ominus})C \leftrightarrow C)) \rightarrow ((\langle \bar{a}\langle b_1 \mathbb{R} \dots b_n \mathbb{R} C \rangle \rangle A) | \langle a[C] \rangle B \rightarrow \langle \tau \rangle b_1 \mathbb{R} \dots b_n \mathbb{R} (A | B))$.

Suppose $P \in [(\langle \bar{a}\langle b_1 \mathbb{R} \dots b_n \mathbb{R} C \rangle \rangle A) | \langle a[C] \rangle B]$, then $P \equiv P_1 | P_2$, $P_1 \xrightarrow{(\nu b_1, \dots, b_n) \bar{a}\langle Q \rangle} P'_1$, $P_2 \xrightarrow{a\langle Q \rangle} P'_2$, $P'_1 \in [[A]]$, $P'_2 \in [[B]]$ and $Q \in [[C]]$. Since $(\ominus b_1, \dots, \ominus b_n)B \leftrightarrow B$, $\{b_1, \dots, b_n\} \cap fn(P'_2) = \emptyset$. Therefore we have $P_1 | P_2 \xrightarrow{\tau} (\nu b_1, \dots, b_n)(P'_1 | P'_2)$. Hence $((\ominus b_1, \dots, \ominus b_n)B \leftrightarrow B) \wedge ((\tilde{\ominus})C \leftrightarrow C)) \rightarrow ((\langle \bar{a}\langle b_1 \mathbb{R} \dots b_n \mathbb{R} C \rangle \rangle A) | \langle a[C] \rangle B \rightarrow \langle \tau \rangle b_1 \mathbb{R} \dots b_n \mathbb{R} (A | B))$.

Case (7): Axiom $(\wedge_{i=1}^n a \neq b_i \wedge a \neq c \wedge ((\ominus a)B \leftrightarrow B) \wedge ((\tilde{\ominus})B \leftrightarrow B)) \rightarrow (a \mathbb{R} \langle \bar{c}\langle b_1 \mathbb{R} \dots b_n \mathbb{R} B \rangle \rangle A \rightarrow \langle \bar{c}\langle b_1 \mathbb{R} \dots b_n \mathbb{R} B \rangle \rangle a \mathbb{R} A)$.

Suppose $P \in [a \mathbb{R} \langle \bar{c}\langle b_1 \mathbb{R} \dots b_n \mathbb{R} B \rangle \rangle A]$, then $P \equiv (\nu a)P_1$, $P_1 \xrightarrow{(\nu b_1, \dots, b_n) \bar{c}\langle Q \rangle} P'_1$, $Q \in [[B]]$, $P'_1 \in [[A]]$. Since $\wedge_{i=1}^n a \neq b_i \wedge a \neq c \wedge ((\ominus a)B \leftrightarrow B) \wedge ((\tilde{\ominus})B \leftrightarrow B)$, $a \notin n(Q)$. Therefore we have $P \equiv (\nu a)P_1 \xrightarrow{(\nu b_1, \dots, b_n) \bar{c}\langle Q \rangle} (\nu a)P'_1$. Hence $(\wedge_{i=1}^n a \neq b_i \wedge a \neq c \wedge ((\ominus a)B \leftrightarrow B) \wedge ((\tilde{\ominus})B \leftrightarrow B)) \rightarrow (a \mathbb{R} \langle \bar{c}\langle b_1 \mathbb{R} \dots b_n \mathbb{R} B \rangle \rangle A \rightarrow \langle \bar{c}\langle b_1 \mathbb{R} \dots b_n \mathbb{R} B \rangle \rangle a \mathbb{R} A)$.

Case (8): Axiom $(a \neq b \wedge \wedge_{i=1}^n b \neq c_i \wedge (B \rightarrow \neg(\ominus b)\top) \wedge ((\tilde{\ominus})B \leftrightarrow B)) \rightarrow (b \mathbb{R} \langle \bar{a}\langle c_1 \mathbb{R} \dots c_n \mathbb{R} B \rangle \rangle A \rightarrow \langle \bar{a}\langle b \mathbb{R} c_1 \mathbb{R} \dots c_n \mathbb{R} B \rangle \rangle A)$.

Suppose $P \in [b \mathbb{R} \langle \bar{a}\langle c_1 \mathbb{R} \dots c_n \mathbb{R} B \rangle \rangle A]$, then $P \equiv (\nu b)P_1$, $P_1 \xrightarrow{(\nu c_1, \dots, c_n) \bar{a}\langle Q \rangle} P'_1$, $Q \in [[B]]$, $P'_1 \in [[A]]$. Since $a \neq b \wedge \wedge_{i=1}^n b \neq c_i \wedge (B \rightarrow \neg(\ominus b)\top) \wedge ((\tilde{\ominus})B \leftrightarrow B)$, $b \in fn(Q)$. Therefore we have $P \equiv (\nu b)P_1 \xrightarrow{(\nu b)(\nu c_1, \dots, c_n) \bar{a}\langle Q \rangle} P'_1$. Hence $(a \neq b \wedge \wedge_{i=1}^n b \neq c_i \wedge (B \rightarrow \neg(\ominus b)\top) \wedge ((\tilde{\ominus})B \leftrightarrow B)) \rightarrow (b \mathbb{R} \langle \bar{a}\langle c_1 \mathbb{R} \dots c_n \mathbb{R} B \rangle \rangle A \rightarrow \langle \bar{a}\langle b \mathbb{R} c_1 \mathbb{R} \dots c_n \mathbb{R} B \rangle \rangle A)$.

Appendix B. Proof of Proposition 2

Proposition 2 There is no finite inference system such that $\Gamma \models_{SL} A \Rightarrow \Gamma \vdash_{SL} A$.

Proof. Let $\Phi = \{\bar{a}\langle 0 \rangle.\top, \bar{a}\langle 0 \rangle.\bar{a}\langle b.0 \rangle.\top, \bar{a}\langle 0 \rangle.\bar{a}\langle b.0 \rangle.\bar{a}\langle b.b.0 \rangle.\top, \bar{a}\langle 0 \rangle.\bar{a}\langle b.0 \rangle.\bar{a}\langle b.b.0 \rangle.\bar{a}\langle b.b.b.0 \rangle.\top, \dots\}$. It is easy to see that any finite subset of Φ can be satisfied in Pr , but Φ can not be satisfied in Pr . Suppose it is not true, let P satisfies Φ . By Lemma 1, there exists n , such that $d(P) = n$. But for any n , there exists φ_n in Φ such that for any P satisfying φ_n , $d(P) > n$. This contradicts the assumption. Therefore Φ can not be satisfied in Pr .

Suppose there is a finite inference system such that $\Gamma \models_{SL} A \Rightarrow \Gamma \vdash_{SL} A$. Since Φ can not be satisfied in Pr , we have $\Phi \models_{SL} \perp$. By the assumption, $\Phi \vdash_{SL} \perp$. Hence there is a proof from Φ to \perp in SL . Since proof is a finite formula sequence, there is finite many formulas φ_i in Φ occur in the proof. Therefore we have $\wedge \Phi_i \vdash_{SL} \perp$, where $\Phi_i = \{\varphi_i \mid \varphi_i \text{ is in the proof}\}$. Then by the soundness of inference system of SL , we have that Φ_i is not satisfiable. Since Φ_i is a finite subset of Φ , this contradicts the assumption. Therefore SL have no finite complete inference system.

Appendix C. Proof of Proposition 3

Proposition 3 For any $P, Q \in Pr^c$, $P \equiv Q \Leftrightarrow P \models_{SL} T^{PS}(Q)$ and $Q \models_{SL} T^{PS}(P) \Leftrightarrow T^{PS}(P) \vdash_{SL} T^{PS}(Q)$ and $T^{PS}(Q) \vdash_{SL} T^{PS}(P)$.

Proof. It is trivial by the definition that $P \equiv Q \Leftrightarrow P \models_{SL} T^{PS}(Q)$ and $Q \models_{SL} T^{PS}(P)$. By the soundness, $T^{PS}(P) \vdash_{SL} T^{PS}(Q) \Rightarrow P \models_{SL} T^{PS}(Q)$. We only need to prove $P \equiv Q \Rightarrow T^{PS}(P) \vdash_{SL} T^{PS}(Q)$ and $T^{PS}(Q) \vdash_{SL} T^{PS}(P)$.

We only discuss the following cases, other cases are similar or trivial:

Case (1): $(\nu m)(\nu n)P \equiv (\nu n)(\nu m)P$: Since $m \mathbb{R} n \mathbb{R} T^{PS}(P) \leftrightarrow n \mathbb{R} m \mathbb{R} T^{PS}(P)$, we have $m \mathbb{R} n \mathbb{R} T^{PS}(P) \vdash_{SL} n \mathbb{R} m \mathbb{R} T^{PS}(P)$. The inverse case is similar.

Case (2): $(\nu a)(P|Q) \equiv P|(\nu a)Q$ if $a \notin fn(P)$: Since $a \notin fn(P)$, $(\ominus a)T^{PS}(P) \leftrightarrow T^{PS}(P)$. Furthermore, since $a \mathbb{R} ((\ominus a)T^{PS}(P)|T^{PS}(Q)) \leftrightarrow (\ominus a)T^{PS}(P)|a \mathbb{R} T^{PS}(Q)$, we have $a \mathbb{R} (T^{PS}(P)|T^{PS}(Q)) \vdash_{SL} T^{PS}(P)|a \mathbb{R} T^{PS}(Q)$. The inverse case is similar.

Appendix D. Proof of Proposition 4

Proposition 4 For any $P, Q \in Pr^c$, $P \xrightarrow{\alpha} Q \Leftrightarrow P \models_{SL} \langle \alpha \rangle T^{PS}(Q) \Leftrightarrow T^{PS}(P) \vdash_{SL} \langle \alpha \rangle T^{PS}(Q)$.

Proof. It is trivial by the definition that $P \xrightarrow{\alpha} Q \Leftrightarrow P \models_{SL} \langle \alpha \rangle T^{PS}(Q)$. By the soundness, $T^{PS}(P) \vdash_{SL} \langle \alpha \rangle T^{PS}(Q) \Rightarrow P \models_{SL} \langle \alpha \rangle T^{PS}(Q)$. We only need to prove $P \xrightarrow{\alpha} Q \Rightarrow P \vdash_{SL} \langle \alpha \rangle T^{PS}(P)$.

We apply the induction on the length of the inference tree of $P \xrightarrow{\alpha} Q$:

Case (1): if the length is 0, then $P \xrightarrow{\alpha} Q$ is in the form of $\bar{a}\langle E \rangle.K \xrightarrow{\bar{a}\langle E \rangle} K$ or $a(U).K \xrightarrow{a\langle E \rangle} K\{E/U\}$.

Subcase (a): $\bar{a}\langle E \rangle.K \xrightarrow{\bar{a}\langle E \rangle} K$: Since $\bar{a}\langle E \rangle.T^{PS}(K) \rightarrow \langle \bar{a}\langle E \rangle \rangle T^{PS}(K)$, we have $\bar{a}\langle E \rangle.T^{PS}(K) \vdash_{SL} \langle \bar{a}\langle E \rangle \rangle T^{PS}(K)$.

Subcase (b): $a(U).K \xrightarrow{a\langle E \rangle} K\{E/U\}$: Since $(a(U).T^{PS}(K) \wedge ((\tilde{\ominus})T^{PS}(E) \leftrightarrow T^{PS}(E))) \rightarrow \langle a[T^{PS}(E)] \rangle T^{PS}(K)\{T^{PS}(E)/U\}$, we have $a(U).T^{PS}(K) \vdash_{SL} \langle a[T^{PS}(E)] \rangle T^{PS}(K)\{T^{PS}(E)/U\}$.

Case (2): Assume the claim holds if length is n , now we discuss the case that length is $n + 1$.

Subcase (a): $\frac{M \xrightarrow{(\nu \tilde{b})\bar{a}\langle E \rangle} M' \quad N \xrightarrow{a\langle E \rangle} N' \quad \tilde{b} \cap fn(N) = \emptyset}{M|N \xrightarrow{\tau} (\nu \tilde{b})(M'|N')} \tilde{b} \cap fn(N) = \emptyset$.

Since $M \xrightarrow{(\nu \tilde{b})\bar{a}\langle E \rangle} M'$, $N \xrightarrow{a\langle E \rangle} N'$, and $\tilde{b} \cap fn(N) = \emptyset$, we have $T^{PS}(M) \rightarrow \langle \bar{a}\langle \tilde{b} \mathbb{R} T^{PS}(E) \rangle \rangle T^{PS}(M')$, $T^{PS}(N) \rightarrow \langle a[T^{PS}(E)] \rangle T^{PS}(N')$ and $(\ominus b_1, \dots, b_n)T^{PS}(E) \leftrightarrow T^{PS}(E)$. By the axiom: $((\ominus b_1, \dots, b_n)T^{PS}(N) \leftrightarrow T^{PS}(N)) \wedge ((\tilde{\ominus})T^{PS}(E) \leftrightarrow T^{PS}(E)) \rightarrow ((\langle \bar{a}\langle b_1 \mathbb{R} \dots b_n \mathbb{R} T^{PS}(E) \rangle \rangle T^{PS}(M)) | \langle a[T^{PS}(E)] \rangle T^{PS}(N) \rightarrow \langle \tau \rangle b_1 \mathbb{R} \dots b_n \mathbb{R} (T^{PS}(M) | T^{PS}(N)))$, we have $P \equiv T^{PS}(M)|T^{PS}(N) \vdash_{SL} \langle \tau \rangle b_1 \mathbb{R} \dots b_n \mathbb{R} (T^{PS}(M')|T^{PS}(N'))$.

Subcase (b): $\frac{M \xrightarrow{b\langle E \rangle} M'}{(\nu a)M \xrightarrow{b\langle E \rangle} (\nu a)M'} a \notin n(\alpha)$.

Since $M \xrightarrow{b\langle E \rangle} M'$ and $a \notin n(b\langle E \rangle)$, we have $T^{PS}(M) \rightarrow \langle b\langle T^{PS}(E) \rangle \rangle T^{PS}(M')$ and $((\ominus a)T^{PS}(E) \wedge ((\tilde{\ominus})T^{PS}(E) \leftrightarrow T^{PS}(E)) \leftrightarrow T^{PS}(E))$. By the axiom $(a \neq b \wedge ((\ominus a)T^{PS}(E) \wedge$

$(\tilde{\ominus})T^{PS}(E) \leftrightarrow T^{PS}(E) \rightarrow (a\mathbb{R}\langle b\langle T^{PS}(E) \rangle \rangle T^{PS}(M) \rightarrow \langle b\langle T^{PS}(E) \rangle \rangle a\mathbb{R}T^{PS}(M))$,
we have $T^{PS}(P) = a\mathbb{R}T^{PS}(M) \vdash_{SL} a\mathbb{R}\langle b\langle T^{PS}(E) \rangle \rangle T^{PS}(M) \vdash_{SL}$
 $\langle b\langle T^{PS}(E) \rangle \rangle a\mathbb{R}T^{PS}(M)$.

Subcase (c): $\frac{M \xrightarrow{(\nu\tilde{c})\tilde{a}\langle E \rangle} M'}{(\nu b)M \xrightarrow{(\nu b, \tilde{c})\tilde{a}\langle E \rangle} M'} a \neq b, b \in fn(E) - \tilde{c}$.

Since $M \xrightarrow{(\nu\tilde{c})\tilde{a}\langle E \rangle} M'$ and $a \neq b, b \in fn(E) - \tilde{c}$, we have $T^{PS}(M) \rightarrow$
 $\langle \tilde{a}\langle \tilde{c}\mathbb{R}T^{PS}(E) \rangle \rangle T^{PS}(M')$ and $a \neq b \wedge \bigwedge_{i=1}^n b \neq c_i \wedge (B \rightarrow \neg(\ominus b)\top)$. By the axiom
 $(a \neq b \wedge \bigwedge_{i=1}^n b \neq c_i \wedge (E \rightarrow \neg(\ominus b)\top) \wedge ((\tilde{\ominus})E \leftrightarrow E)) \rightarrow (b\mathbb{R}\langle \tilde{a}\langle c_1\mathbb{R}\dots c_n\mathbb{R}T^{PS}(E) \rangle \rangle$
 $T^{PS}(M') \rightarrow \langle \tilde{a}\langle b\mathbb{R}c_1\mathbb{R}\dots c_n\mathbb{R}T^{PS}(E) \rangle \rangle T^{PS}(M')$, we have $T^{PS}(P) = b\mathbb{R}T^{PS}(M)$
 $\vdash_{SL} (b\mathbb{R}\langle \tilde{a}\langle c_1\mathbb{R}\dots c_n\mathbb{R}T^{PS}(E) \rangle \rangle T^{PS}(M') \vdash_{SL} \langle \tilde{a}\langle b\mathbb{R}c_1\mathbb{R}\dots c_n\mathbb{R}T^{PS}(E) \rangle \rangle T^{PS}(M')$.

Appendix E. Proof of Proposition 11

Proposition 11 For any $A \in WL$, $T^{WM}(A) \in \mu SL$; for any $P \in Pr$,
 $P \models_{\mu SL} T^{WM}(A) \Leftrightarrow P \models_{WL} A$.

Proof : We only discuss the case $A = \langle \langle a\langle A_1 \rangle \rangle \rangle A_2$, other cases are similar.

Suppose $P \models_{\mu SL} T^{WM}(A)$. Since $[[\mu X.C(X)]]_{Pr}^e = \cup_i [[C^i(\perp)]]_{Pr}^e$, if P
 $\in [[\mu X.C(X)]]_{Pr}^e$, then $P \in [[C^i(\perp)]]_{Pr}^e$ for some i . Let $B = \langle a\langle T^{WM}(A_1) \rangle \rangle$
 $(\mu Y.(T^{WM}(A_2) \vee \langle \tau \rangle Y))$, then $P \models_{\mu SL} B \vee \langle \tau \rangle B \vee \langle \tau \rangle \langle \tau \rangle B \dots \vee \langle \tau \rangle^i B$, here
 $\langle \tau \rangle^{i+1} B$ denotes $\langle \tau \rangle(\langle \tau \rangle^i B)$, $\langle \tau \rangle^0 B$ is B . Hence $P \xRightarrow{\varepsilon} Q$, $Q \in [[\langle a\langle T^{WM}(A_1) \rangle \rangle$
 $(\mu Y.(T^{WM}(A_2) \vee \langle \tau \rangle Y))]]_{Pr}^e$. Hence $Q \xrightarrow{a\langle E \rangle} Q'$, $E \in [[T^{WM}(A_1)]]_{Pr}^e$, and $Q' \in$
 $[[\mu Y.(T^{WM}(A_2) \vee \langle \tau \rangle Y)]]_{Pr}^e$. By the similar discuss, we have that $Q' \xRightarrow{\varepsilon} Q''$
and $Q'' \in [[T^{WM}(A_2)]]_{Pr}^e$. Hence $P \xRightarrow{a\langle E \rangle} Q''$, $E \in [[T^{WM}(A_1)]]_{Pr}^e$, and $Q'' \in$
 $[[T^{WM}(A_2)]]_{Pr}^e$. We have $P \models_{WL} A$. The converse claim is similar.

Appendix F. Proof of Proposition 12

Proposition 12 $\vdash_{\mu SL} A!A \leftrightarrow !A$

Proof : Since by the inference system, $\vdash_{\mu SL} S(\mu X.S(X)) \rightarrow \mu X.S(X)$, we
have $\neg \mu X.S(X) \rightarrow \neg S(\mu X.S(X))$. Let $S(X) = \neg(A|\neg X)$, then $\neg \mu X.S(X) =$
 $\neg \mu X.\neg(A|\neg X) = !A$, $\neg S(\mu X.S(X)) = A|\neg \mu X.\neg(A|\neg X) = A!A$. Therefore we
get $\vdash_{\mu SL} !A \rightarrow A!A$.

Since by the inference system, $\vdash_{\mu SL} !A \rightarrow A!A$, we have $\vdash_{\mu SL} \neg(A|A!A) \rightarrow$
 $\neg(A!A)$. Let $T(X) = \neg(A|\neg X)$, then $T(\neg(A!A)) = \neg(A|A!A)$. Since $\vdash_{\mu SL}$
 $T(\neg(A!A)) \rightarrow \neg(A!A)$, by the inference system, we have $\vdash_{\mu SL} \mu X.T(X) \rightarrow$
 $\neg(A!A)$. Furthermore, $\mu X.T(X) = \mu X.\neg(A|\neg X) = \neg !A$, hence $\vdash_{\mu SL} \neg !A \rightarrow$
 $\neg(A!A)$, we have $\vdash_{\mu SL} A!A \rightarrow !A$.